Homework 2: Intro to R for Epidemiologists

Instructions

- Due at 11:59 PM on Wednesday, March 4, 2015
- Late assignments will not be accepted
- Covers topics in weeks 4 - 6
- Submit this assignment as a .R file by e-mail to jenna.krall@emory.edu
- Name your file as "LASTNAME_560R_hw2.R"
- **Before submitting the assignment, comment out any `rm(list = ls())`, `setwd()`, or specific file paths to your computer (e.g. in `read.csv()`)**

You may work with your classmates, but you must type and turn in your own assignment. Do not copy and paste someone else's work. Your code comments should be your own.

Grading

The assignment is worth 15% of your final grade and will be graded out of 30 total points. For this assignment, you may make sure your Homework 2 will be fully graded by setting your working directory to the directory where your code is located and running the following commands:

```
install("devtools")
library(devtools)
install_github("homework560R", "kralljr")
library(homework560R)
rm(list = ls())
grade_homework2("Krall_560R_hw2.R")
```

- **You will not receive any credit for code that generates errors.**

Additionally, you will not receive credit for solutions such as:

```
mean_diab <- 5.2
```

You must, whenever possible, use functions on the right hand side:

```
mean_diab <- mean(diabetes)
```

1 Part 0. Code formatting and commenting (2 points)

2 Part 1. Sleep duration recommendations (8 points)

1. The National Sleep Foundation recently updated its recommendations for sleep duration by age group. Using the information provided at http://sleepfoundation.org/sites/default/files/STREPchanges_1.png, write an `if/then/else` statement to return the sleep duration recommendation **only for children ages 1 -13 years**.
    - Your `if/then/else` statement should have an input R object `age_kid` that defines the age of the child.

- Within your `if/then/else` statement, you should define a variable `sleep_duration` based off of `age_kid` that is one of: "11-14 hours", "10-13 hours", "9-11 hours", and "not defined". Be sure these values are transcribed exactly.
- Your `if/then/else` statement should not print any results.
- You must comment out any assignment values `age_kid` that you make (e.g. `age_kid <- 2`).

3 Part 2. Diabetes (20 points)

In this section, you will be summarizing a dataset using a table. This dataset contains 403 individuals screened for diabetes.

References

- Diabetes: http://www.cdc.gov/diabetes/home/
- Data accessed February 6, 2015, http://biostat.mc.vanderbilt.edu/wiki/Main/DataSets.

In this dataset, there are missing values. Do not remove any missing values from the entire dataset (e.g. using `complete.cases`), and instead use arguments within functions (e.g. `na.rm`) to remove missing values for computing statistics such as means.

1. Read in the datasets `diabetes_female.csv` and `diabetes_male.csv`.
2. Merge these two datasets together to create a new R data frame called `diabetes`.
3. Use one apply statement to find the overall mean for each of total cholesterol, hdl cholesterol, age, height, and weight.
4. Add a new variable to your dataset called `diab1` that is 1 for individuals with diabetes and 0 for individuals without diabetes, where diabetes is defined as having glycosolated hemoglobin > 7.
5. Create a grouped dataset that is grouped by diabetes status.
6. Using your grouped dataset, create a new data frame called `tab_diab`.
   (a) Compute the mean of total cholesterol, hdl cholesterol, age, height, and weight (by diabetes status) using one line of code.
   (b) Clean this matrix by removing any rows corresponding to `NA` and possibly transposing the matrix so the rows correspond to the variables (Hint: `?t`).
   (c) Convert your final matrix to a dataframe using the function `data.frame`.
7. Do the same thing in (6.) computing the number of non-missing observations for each variable by diabetes status. Add these new variables to `tab_diab`.
8. Add overall the means from (3.) to `tab_diab` as a new variable called `Mean`.
9. Add the differences in means (`Mean_diff`) for each variable between diabetics and not diabetics to `tab_diab`.
10. We want to test whether the differences in means between diabetics and non-diabetics are statistically significantly different than zero for total cholesterol, hdl cholesterol, age, height, and weight. We can use t-tests to do this.
    (a) Write a "for loop" that loops over variables,
    (b) For each variable, run a t-test comparing diabetics to non-diabetics.

(c) Save the p-value, lower confidence bound, and upper confidence bounds (corresponding to the 95% confidence interval) in new vectors.

(d) Add your new vectors for p-value, lower confidence bound, and upper confidence bounds to `tab_diab`.

11. Your final data frame, `tab_diab`, should look EXACTLY as below, but with values filled in.

```
##   Variable Mean N_diabetic Mean_diabetic N_not_diabetic Mean_not_diabetic
## 1     chol   --         --            --             --                --
## 2      hdl   --         --            --             --                --
## 3      age   --         --            --             --                --
## 4   height   --         --            --             --                --
## 5   weight   --         --            --             --                --
##   Meandiff Meandiff_LB Meandiff_UB Meandiff_Pval
## 1       --          --          --            --
## 2       --          --          --            --
## 3       --          --          --            --
## 4       --          --          --            --
## 5       --          --          --            --
```

4 Bonus (1 - 5 points)

Create a plot of your choosing using the diabetes dataset.